

SIMULASI ALAT NOTIFIKASI SERVIS RINGAN SEPEDA MOTOR OTOMATIS BERBASIS *ARDUINO NANO* TERINTEGRASI DENGAN *ANDROID*

Ratih Ayuninghemi¹, Mochammad Faqih Abdillah²

^{1,2} Jurusan Teknologi Informasi, Politeknik Negeri Jember
¹ratihayuninghemi@gmail.com, ²mfaqih.abdillah@gmail.com

Abstrak

Pemakaian sepeda motor di Indonesia sebagai mode transportasi pribadi menjadi kebutuhan pokok, kecenderungan pemakaian sepeda motor secara terus menerus terkadang melupakan performa mesin, yang membutuhkan servis ringan (*tune up*) secara periodik, ini menjadi sangat penting ketika melihat performa mesin menurun dan beberapa *sparepart* atau oli mesin juga terdapat masa pakai, oleh karenanya pentingnya ada sebuah alat untuk mengingatkan pengendara sepeda motor untuk servis secara berkala. Untuk itu pembuatan simulasi alat notifikasi servis ringan sepeda motor otomatis ini bertujuan untuk membuat alat notifikasi servis ringan berupa notifikasi mengganti oli dengan jarak waktu pemakaian (bulan) dan jarak tempuh kendaraan, yang terkoneksi dengan *smartphone* pengguna (pengendara), penelitian ini menghasilkan alat hitung digital (*speedometer android*) yang membaca data dari *arduino nano* sebagai mikrokontroler dan terhubung secara *wireless* dengan *Bluetooth*, alat ini diuji dengan 2 sepeda motor yang berbeda tipe jarak pembacaan jarum pada *speedometer*nya, percobaan pertama menggunakan sepeda motor mio sporty yang mendapatkan hasil notifikasi muncul pada saat mencapai batas kilometer tercapai tetapi terdapat perbedaan hasil perhitungan jarak antara *speedometer* dan *speedometer android*, percobaan kedua menggunakan sepeda motor mio jet fi yang mendapatkan hasil notifikasi muncul pada saat batas kilometer telah tercapai dan hasil cukup akurat. Dan percobaan dengan jarak bulan waktu servis yang mendapatkan hasil akurat.

Kata kunci : *Android, Arduino, Speedometer, Sepeda Motor, Servis Ringan, Smartphone.*

1. Pendahuluan

Realita dalam masyarakat Indonesia bahwa pemakaian sepeda motor setiap harinya telah menjadi kebutuhan pokok, misalnya untuk mobilitas dari satu tempat ke tempat lainnya. Kecenderungan memakai sepeda motor membuat pengendara terkadang melupakan tentang kondisi dari sepeda motor itu sendiri, hal ini sangatlah tidak relevan ketika pemakaian terus menerus tetapi tidak mengecek kondisi sepeda motor yang bisa saja mengalami kerusakan sewaktu-waktu. Cara yang mudah untuk mengecek kondisi motor adalah dengan servis ringan atau hanya sekedar *tune up*, seperti yang diketahui bahwa servis ringan atau *tune up* tidaklah terlalu sulit untuk dilakukan oleh seorang awam, *tune up* ataupun servis ringan bisa dilakukan dengan cara mengganti oli mesin sepeda motor pada batas kilometer tertentu contohnya 2000 KM ataupun dalam jangka waktu tertentu contohnya 2 bulan. Jika saja ada alat yang dapat membantu memberikan notifikasi kepada pengendara sepeda motor untuk servis ringan atau *tune up* mungkin akan lebih memudahkan dan juga sepeda motor akan tetap dalam kondisi yang prima.

Ada beberapa alat yang dapat membantu pengendara untuk mendapatkan notifikasi servis yang telah diciptakan sebelumnya, seperti (1)

aplikasi “*My Service Reminder*” berbasis *android*. Aplikasi ini mampu memberikan peringatan kapan waktu servis dan pergantian komponen kendaraan dilakukan dengan melakukan *input* data saat mulai menggunakan aplikasi. Sehingga pengguna sekaligus pengendara dapat selalu menjaga kendaraannya tetap aman dan nyaman dalam berkendara (Soni Dwi Sandi, 2013). (2) Aplikasi “*Paduan Tune Up Ringan Sepeda Motor 4 Tak melalui Handphone menggunakan J2ME*” yang dapat menyajikan panduan dalam mendeteksi kerusakan motor 4 tak yang dapat dialami oleh para pengguna sepeda motor beserta cara penanggulangannya, dengan profil MIDP 1.0 pengguna lebih rileks dalam menjalankan aplikasi ini dalam mengatasi masalah pada sepeda motornya (Muhamamd Firdaus, 2009).

Melihat masalah yang terjadi diatas penulis membuat simulasi alat notifikasi servis ringan (*tune up*) sepeda motor otomatis berbasis *arduino nano* terintegrasi dengan *android*. Alat ini bekerja di *platform android* melihat banyaknya pengguna *android* di Indonesia yang akan memudahkan pengguna (*user*) untuk mengetahui kapan penggantian harus dilakukan untuk sepeda motor masing – masing. Kelebihan alat ini dapat mengatur jarak batas tempuh dari *speedometer* (KM/h) untuk mendapatkan notifikasi penggantian oli mesin, tetapi tidak pada komponen atau *sparepart* yang lain, dan

juga alat ini bekerja secara *realtime* pada saat dikendarai. Alat ini memiliki beberapa fitur diantaranya: (1) Memberikan informasi tentang tempuh *speedometer* dalam satu kali perjalanan, (2) pada saat mencapai batas limit kilometer yang ditentukan, maka akan ada notifikasi untuk pengguna (pengendara sepeda motor) bahwa speedometer telah mencapai batas penggantian oli, (3) tidak hanya jarak tempuh pada *speedometer* akan tetapi perhitungan jarak pergantian oli dengan melihat tanggal penggantian oli atau menggunakan interval bulan. Terintegrasinya dengan *Android* membuat alat ini mudah untuk digunakan (*user friendly*) via *bluetooth* yang digunakan oleh *Arduino* sendiri akan mempermudah pengecekan *speedometer* untuk dilakukan servis ringan atau penggantian oli secara periodik.

2. Metode Pengembangan Alat

Dalam penelitian “Simulasi Alat Notifikasi Servis Ringan Sepeda Motor Otomatis Berbasis *Arduino nano* Terintegrasi dengan *Android*” ini menggunakan tahapan metode pelaksanaan yaitu analisa kebutuhan, perancangan, pembuatan, pengujian, analisa data.

2.1. Analisa Kebutuhan

Berbagai Referensi didapatkan dari Buku Pustaka, Penelitian yang telah ada sebelumnya serta melalui media *internet* untuk mengumpulkan data dan informasi yang diperlukan untuk analisis kebutuhan untuk pelaksanaan kegiatan tugas akhir ini. Penulis melakukan percobaan terhadap *Arduino*, sebagai mikrokontroler untuk membaca putaran roda yang dibantu dengan *Reed Switch sensor*. *Arduino* dan *Bluetooth HC-05* modul juga diperlukan untuk mengkoneksikan terhadap *smartphone* pengguna (pengendara sepeda motor) agar dapat memonitoring dan mendapatkan notifikasi pada saat mencapai batas limitasi kilometer.

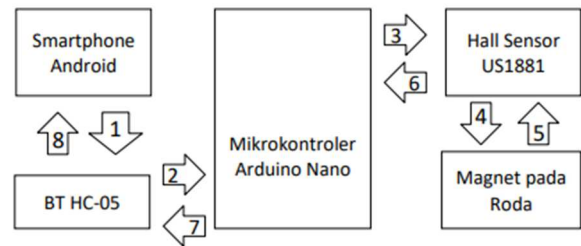
2.2. Perancangan

Pada tahap ini perancangan dibagi menjadi dua yaitu perancangan perangkat keras untuk alat dan perancangan perangkat lunak untuk aplikasi.

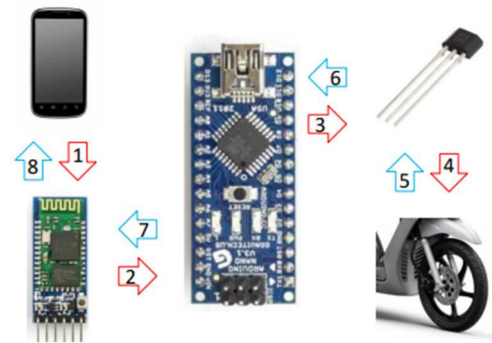
2.2.1. Perancangan Perangkat Keras

Pada tahap ini, dilakukan perancangan alat kontrol yang sesuai dengan perencanaan. Perangkat kontrol tersebut terdiri dari sebuah *Smartphone* sebagai alat input dan monitoring, *Arduino* sebagai alat pengontrol, *Reed Swtich Sensor* sebagai penghubung antara *Arduino* dan *Magnet Roda*, *Bluetooth HC-05* sebagai media penghubung antara *android smartphone* dengan *Arduino*. Adapun

perancangan dari perangkat keras itu sendiri terdapat pada gambar berikut.



Gambar 1. Blok diagram perancangan perangkat keras



Gambar 1. Ilustrasi blok diagram perancangan perangkat keras

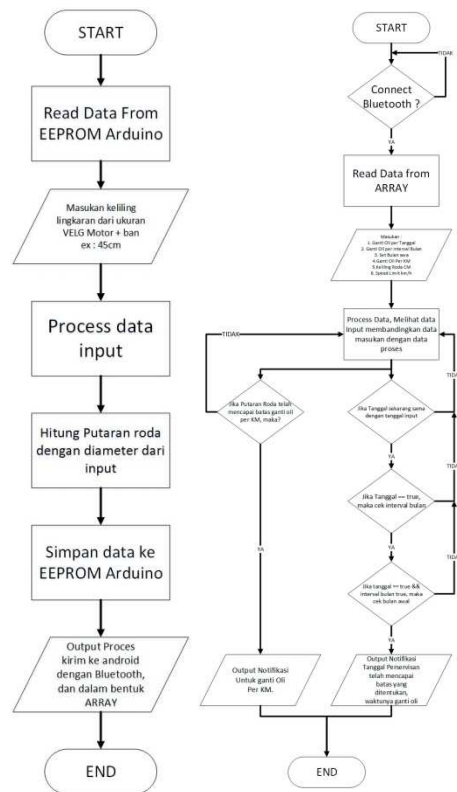
2.2.2. Perancangan Perangkat Lunak

Pada tahap ini, dilakukan perancangan pada aplikasi kontrol dan monitoring berbasis *android* yang sesuai dengan perencanaan. Dimana pada perancangan ini dibuat sebuah skema untuk pembuatan aplikasi, sehingga aplikasi tersebut dapat menjadi media kontrol dan jembatan dari media yang akan dikontrol dan display menggunakan aplikasi ini. Adapun alur pembuatan program itu sendiri terdapat pada gambar 3 berikut.



Gambar 3. Ilustrasi blok diagram perancangan perangkat lunak

Kemudian setelah melakukan perancangan terhadap program, selanjutnya maka perancangan dari flowchart program itu sendiri. Sehingga program dapat direncanakan ingin dibuat seperti bagaimana dan sesuai dengan kebutuhan. Adapun flowchart program itu sendiri terdapat pada gambar 4 berikut.



Gambar 4. Flowchart program Arduino dan Android

2.3. Pembuatan

Pada tahapan ini penulis melakukan pembuatan alat pembaca putaran roda, dimana dalam pembuatan alat kontrol ini penulis tidak seratus persen membuat perangkat keras melainkan perangkat keras sudah jadi (Arduino, BT HC-05, Reed Switch) hal ini dikarenakan penulis melakukan pengembangan terhadap tugas akhir terdahulu yang mana pada saran terdapat “pengembangan aplikasi yang realtime dengan penambahan perangkat keras” sehingga penulis membuat konfigurasi baru dan penambahan perangkat keras tersebut sehingga dapat berkomunikasi dan bekerja sebagai mana mestinya.

Setelah membuat perangkat keras penulis memulai untuk pembuatan perangkat lunak atau aplikasi monitoring serta kontrolnya untuk alat tersebut, dengan menggunakan Arduino IDE sebagai software pemrograman mikrokontroler dan Android Studio sebagai software pembuat aplikasi android yang nantinya berbentuk apk.

2.4. Pengujian

Pada tahapan ini penulis melakukan pengujian terhadap simulasi alat notifikasi servis ringan sepeda motor otomatis berbasis arduino nano terintegrasi dengan android menggunakan *smartphone* android, langkah pertama ialah melakukan pengecekan terhadap semua kabel jumper apakah sudah terkoneksi dengan benar atau tidak, kemudian menghidupkan arduino dengan memencet tombol saklar dan cek apakah sudah turned on dengan

indikator led pada masing masing hardware, selanjutnya mengkoneksikan bluetooth HC-05 dengan bluetooth *smartphone*., langkah selanjutnya memberikan inputan batas limitasi kilometer kepada alat dengan aplikasi yang ada pada android *smartphone*. Langkah terakhir adalah dengan mengendarai sepeda motor dan mengamati melalui *smartphone* apakah kilometer pada display handphone berjalan semestinya dan sesuai pada jarak batas yang ditentukan.

2.5. Analisa Data

Pada bagian ini akan mengumpulkan berbagai masalah dari apa yang terjadi selama pengujian. Memperbaiki segala kekurangan dari alat dan melakukan pengujian hingga didapatkan sesuatu alat yang sesuai dengan harapan dan tujuan.

3. Hasil Dan Pembahasan

Bagian ini adalah membahas tentang bagaimana hasil uji coba yang telah dilakukan setelah alat dan aplikasi android telah selesai diimplementasikan ke sepeda motor, uji coba dilakukan dengan 2 tipe sepeda motor yang memiliki perbedaan tipe speedometer dan jarak pembacaan jarum speedometer, yaitu mio sporty dan mio jet fi, serta pengujian untuk jarak waktu servis yaitu dengan satuan bulan.

3.1. Hasil Pembuatan Perangkat Keras

Hasil pembuatan alat secara keseluruhan yang meliputi *Arduino nano*, *Bluetooth HC-05*, *Regulator 7805*, *DC JACK Barrel*, *Electrolit Condensator (Elco) 16V 4.7pF* menjadi satu perangkat keras modul dan Sensor *Reed Switch*. Berikut gambar alat keseluruhan.



Gambar 5. Rangkaian Alat secara keseluruhan

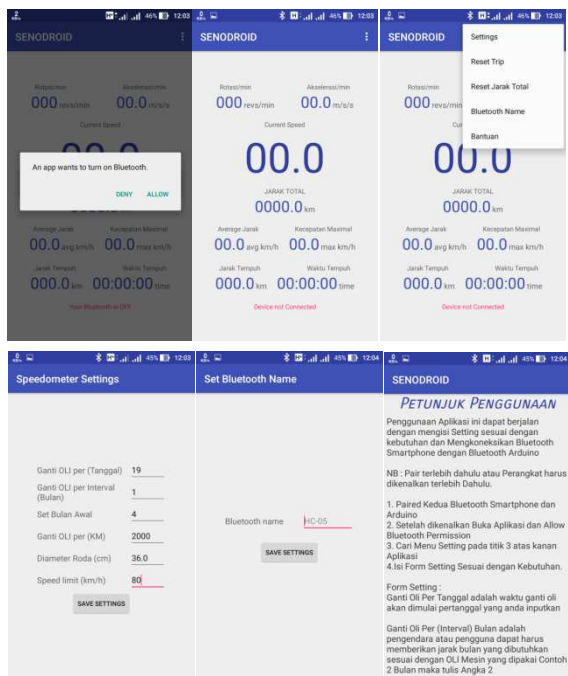
Module Sensor Reed Switch mengirim data masuk atau data *input* pada *arduino nano* melalui *digital pin 5*, setelah data diolah dan telah menghasilkan *outputan* maka *arduino nano* mengirim hasil olahan data ke *smartphone* user yang telah terinstall aplikasi dan terkoneksi melalui wireless module *bluetooth HC-05*. Maka akan muncul semacam monitoring digital via *smartphone* layaknya *speedometer* dan apabila telah mencapai batas limit yang telah ditentukan oleh pengguna maka notifikasi secara otomatis akan keluar pada

statusbar dengan *vibration*/getar pada smartphone pengguna.

3.2. Hasil Pembuatan Aplikasi

Pada tahap ini pembuatan aplikasi android menggunakan *Android Studio*, dimana kebutuhan fungsi dari fitur *android studio* lebih kompleks dari software lainnya.

Setelah perancangan fitur yang dimasukkan dan penulis berhasil membuat alat untuk *monitoring* yaitu seperti *speedometer digital* dengan aplikasi android. Dengan beberapa masukan serta kebutuhan fitur yang tidak hanya menu utama pada aplikasi, penulis menambahkan fitur *setting*, reset trip, reset total perjalanan, nama *bluetooth* dan bantuan, desain *interface* seperti ditunjukkan pada gambar berikut.



Gambar 6. Interface Aplikasi Android

3.3. Hasil Pengujian Alat

Pada tahap pengujian alat dipastikan telah dapat berkomunikasi dengan baik, antara dua *device* yaitu android dan alat yang dipasang pada sepeda motor.

3.3.1. Pengujian menggunakan Mio Sporty

Dalam pengujian Alat menggunakan sepeda motor mio sporty penulis mendapatkan hasil bahwa terdapat perbedaan pembacaan kecepatan putaran roda, dengan selisih 10 angka antara *speedometer* pada sepeda motor dan *speedometer* pada *android*.

Pengujian alat pertama dengan jarak 1KM untuk mendapatkan notifikasi pada *android*, dengan jarak awal *speedometer* 39332,0 lalu mendapatkan notifikasi pada jarak tempuh 39333,1 dan pada *speedometer android* jarak tempuh awal 1,0 dan

mendapatkan notifikasi pada jarak tempuh 2,0. Berikut gambar 4.21 notifikasi muncul pada android dengan jarak yang telah ditempuh.

Pengujian Alat kedua kalinya dengan jarak 2KM untuk mendapatkan notifikasi pada *android*, dengan jarak awal *speedometer* 39333,1 lalu memperoleh notifikasi pada jarak tempuh 39335,4 dan pada *speedometer android* jarak tempuh awal 2,0 dan mendapatkan notifikasi pada jarak tempuh total 4,0

Dengan dua variabel jarak tempuh pengujian mendapatkan perbedaan jarak yang cukup signifikan dikarenakan pada percobaan kedua penulis mengendarai dengan kecepatan yang tidak stabil seperti percobaan pertama yaitu mulai dengan 20KM/Jam hingga 60KM/Jam hal ini juga yang mempengaruhi alat dan hasil hitung *speedometer* asli berbeda. Berikut Hasil pengujian alat dengan menggunakan motor mio sporty dalam bentuk tabel 1 dan 2.

Tabel 1. Percobaan pertama dengan jarak 1KM

Jenis Sepeda Motor	Mio Sporty	
Interval Jarak	1 Kilometer	
Jarak Sebelum	39332,0 (Speedometer Motor)	1,0 (Speedometer android)
Jarak Sesudah	39333,1 (Speedometer Motor)	2,0 (Speedometer android)
Hasil	Notifikasi Muncul	
Keterangan	Cukup akurat ada selisih 100m pada <i>speedometer</i>	

Tabel 2. Percobaan pertama dengan jarak 2KM

Jenis Sepeda Motor	Mio Sporty	
Interval Jarak	2 Kilometer	
Jarak Sebelum	39333,1 (Speedometer Motor)	2,0 (Speedometer android)
Jarak Sesudah	39335,4 (Speedometer Motor)	4,0 (Speedometer android)
Hasil	Notifikasi Muncul	
Keterangan	Cukup akurat ada selisih 400m pada <i>speedometer</i>	

3.3.2. Pengujian Menggunakan Mio Jet FI

Dalam pengujian alat pertama menggunakan Mio Jet FI penulis mendapatkan hasil bahwa alat dapat bekerja dengan baik, *current speed speedometer* sepeda motor dengan *current speed speedometer* pada *android* tidak jauh berbeda, notifikasi muncul pada jarak interval jarak 1KM dengan jarak awal *speedometer* adalah 15815,0 lalu muncul notifikasi pada jarak tempuh 15816,6 sedangkan pada alat jarak awal 36,6 dan jarak tempuh hasil 37,6. Terjadi perbedaan 600m pada *speedometer* asli dikarenakan *lost connection* antara alat dengan *smartphone* yang disebabkan oleh guncangan pada alat sehingga

module bluetooth tersengol dan otomatis *restarting* pada alat.

Pengujian kedua dilakukan dengan jarak tempuh 2KM dengan jarak awal *speedometer* adalah 15819,5 lalu notifikasi muncul pada jarak tempuh 15821,5 sedangkan pada alat jarak awal 37,9 dan jarak tempuh hasil 39,9 lalu notifikasi muncul. Berikut Hasil pengujian alat dengan menggunakan motor mio jet fi dalam bentuk tabel 3 dan 4.

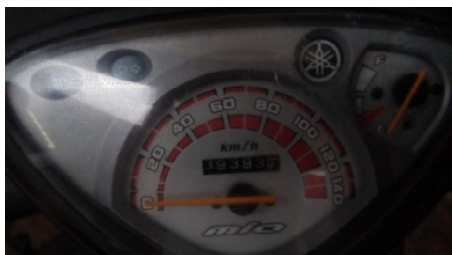
Tabel 3. Percobaan pertama dengan jarak 1KM

Jenis Sepeda Motor	Mio Jet FI	
Interval Jarak	1 Kilometer	
Jarak Sebelum	15815,0 (Speedometer Motor)	36,6 (Speedometer android)
Jarak Sesudah	15816,6(Speedometer Motor)	37,6 (Speedometer android)
Hasil	Notifikasi Muncul	
Keterangan	Akurat tetapi karena guncangan sehingga terjadi lost connection antara android dan arduino	

Tabel 4. Percobaan pertama dengan jarak 2KM

Jenis Sepeda Motor	Mio Jet FI	
Interval Jarak	2 Kilometer	
Jarak Sebelum	15819,5 (Speedometer Motor)	37,9 (Speedometer android)
Jarak Sesudah	15821,5 (Speedometer Motor)	39,9 (Speedometer android)
Hasil	Notifikasi Muncul	
Keterangan	Akurat	

Setelah melakukan dua kali pengujian terhadap dua sepeda motor dengan ring velg 14inch penulis mendapatkan perbedaan antara pengujian alat dengan menggunakan motor yamaha Mio Sporty dan yamaha Mio Jet FI, dikarenakan perbedaan angka penunjuk pada *speedometer*. Adapun perbedaannya adalah seperti pada gambar 7 dan 8 berikut.



Gambar 7. Speedometer Mio Sporty



Gambar 8. Speedometer Mio JET FI

3.4. Pengujian Alat dengan Notifikasi Jarak Bulan Servis

Tabel 5. Notifikasi servis dengan jarak bulan

No	Interval Bulan	Bulan Sebelumnya	Bulan Notifikasi	Hasil
1	1 Bulan	25 April 2018	25 Mei 2018	Akurat
2	2 Bulan	25 Maret 2018	22 Mei 2018	Akurat
3	3 Bulan	22 Februari 2018	27 Mei 2018	Akurat

Dalam perhitungan Bulan ini dapat diketahui bahwa notifikasi muncul setelah set bulan awal yang ditentukan untuk notifikasi oleh pengguna, dimana dengan interval bulan sebenarnya sudah dapat dilihat alat telah berjalan sesuai dengan kebutuhan pengguna.

4. Kesimpulan

Kesimpulan yang dapat diperoleh dari simulasi alat notifikasi servis ringan sepeda motor otomatis berbasis *arduino nano* terintegrasi dengan *android* yaitu:

1. Notifikasi servis pada interval jarak tempuh sepeda motor dalam hal ini menghitung putaran roda dapat berjalan, tetapi mendapatkan perbedaan pada percobaan pertama dengan menggunakan Mio Sporty dan beberapa sepeda motor lainnya, hanya pada sepeda motor Mio Jet FI yang bisa menghasilkan notifikasi cukup akurat dikarenakan perbedaan *speedometer* pada sepeda motor.
2. Notifikasi untuk interval jarak bulan dilakukan percobaan dengan jarak bulan yang mendapatkan hasil akurat dimana notifikasi muncul setelah memasuki hari yang ditentukan.

Daftar Pustaka :

- Abdo, Edward. 2nd edision, Modern Motorcycle Technology, Delmar: Cengage Learning.
Cossalter, Vittore. 2006. Motorcycle Dynamics. Lulu. ISBN 978-1-4303-0861-4.

- Firdaus, Muhammad. 2009, Aplikasi Panduan Tune Up Ringan Sepeda Motor 4 Tak Melalui Handphone Dengan Menggunakan J2ME. Jakarta: Universitas Gunadarma
- Kadir, Abdul. 2015. *From Zero to a pro:Arduino*. Jakarta: Gudang Penerbit
- Manjoo. Farhad. 2015, A Murky Road Ahead for Android, Despite Market Dominance The New York Times.
- Marsudi, M.T. 2010, Teknisi Otodidak Sepeda Motor (Belajar Teknik dan Perawatan Kendaraan Ringan Mesin 4 Tak), Yogyakarta: C.V Andi OFFSET Yogyakarta.
- Nazaruddin Safaat H. 2011. Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung: Informatika
- Nugraha Setya. Beni. 2005, TUNE UP SEPEDA MOTOR. Yogyakarta: Modul Fakultas Teknik Universitas Negeri Yogyakarta.
- Sandi Dwi. Soni. 2013, Aplikasi “My Service Reminder” Berbasis Android. Yogyakarta: Naskah Publikasi Amikom Yogyakarta.